

КУРСОВ ПРОЕКТ

ДИСЦИПЛИНА: ПРОЕКТИРАНЕ И ИНТЕГРИРАНЕ НА СОФТУЕРНИ СИСТЕМИ

РЕАЛИЗАЦИЯ НА СИСТЕМАТА

ВЕРСИЯ 1.0



ФАК. №	ИМЕ НА СТУДЕНТ	СЕКЦИЯ ОТ ДОКУМЕНТА
62445	МАРИЯ ВЕЛЕВА	4, 5
62455	АНДРЕЯ ДЯКОВА	1, 6, 7
62473	Ирина Христова	2, 3

ЯНУАРИ, 2023

1	Въведение.....	3
1.1	Цел	3
1.2	Резюме	3
1.3	Дефиниции и акроними	4
2	Използвани технологии	6
3	Реализация на базата от данни	8
4	Реализация на бизнес логиката	10
5	Реализация на потребителския интерфейс.....	11
5.1	Вход	11
5.2	Регистрация.....	12
5.3	Начална страница	13
5.4	Профил	16
5.5	Messenger.....	17
6	Внедряване на системата	18
7	Разпределение на дейностите по реализацията	19

1 ВЪВЕДЕНИЕ

1.1 Цел

Този документ описва работата по разработката на системата *Worldwide Lens*. Той предоставя информация за използваните техники за създаване на системата, като например кои технологии са използвани и как са внедрени. Той също така включва изображения на потребителския интерфейс и обяснения на всяка функционалност и как да получите достъп до тях.

Целта на документа е да покаже как *Worldwide Lens* може да бъде интегрирана в конкретно устройство. Документът е ръководство за изграждане на системата от нулата и представя реализацията на бизнес логиката, интерфейса и базите данни.

1.2 Резюме

Документът е разделен е на следните няколко секции:

1 Въведение

Изяснява целта на документа и идентифицира кой раздел от него описва конкретна информация и нейното местоположение в документа.

2 Използвани технологии

Описва технологиите и методите, използвани за изпълнение на системата.

3 Реализация на базата от данни

Очертава категоризация и структурата на базата данни.

4 Реализация на бизнес логиката

В тази част на документа се обяснява как е приложена бизнес логиката на системата с помощта на избраните технологии.

5 Реализация на потребителския интерфейс

Очертава въведените елементи на потребителския интерфейс и различните изгледи. Документът включва изображения и описания на всеки изглед, за да осигури визуално представяне на дизайна.

6 Внедряване на системата

Обяснява необходимите действия за внедряване на системата, включително необходимите изисквания, които трябва да бъдат изпълнени.

7 Разпределение на дейностите по реализацията

Описва задачите и отговорностите, които всеки член на екипа изпълнява по време на разработката на системата.

1.3 Дефиниции и акроними

- **HTTP**

Hypertext Transfer Protocol е протокол за предаване на данни по интернет, използван за създаване и достъп до уеб страници и друга информация в *World Wide Web*. Базира се на модел заявка-отговор, при който клиент изпраща заявка до сървър и сървърът връща отговор.

- **REST**

Стил софтуерна архитектура за реализация на уеб услуги. Основната идея е да се определи системен ресурс, който се променя в резултат на взаимодействието между доставчика на услуги и потребителя. Архитектурният модел *REST* включва взаимодействията между сървър и клиент, осъществени по време на трансфера на данни.

- **Rest API**

Representational State Transfer API е уеб стандарт за проектиране и изграждане на уеб услуги, които използват *HTTP* заявки за *POST*, *PUT*, *GET* и *DELETE* данни. Той дефинира *URL* адреси на крайна точка и *HTTP* методи за взаимодействие с ресурси и е прост, гъвкав и мащабируем начин за достъп и манипулиране на уеб базирани услуги.

- **Потребителски интерфейс**

Точката на взаимодействие между потребител и компютърна програма или устройство, включва елементите, с които потребителят взаимодейства, като бутони, менюта и текстови полета. Дизайнът на потребителския интерфейс има за цел да направи програмата или устройството лесни за използване и разбиране, като предоставя ясна и организирана информация и опции, визуално привлекателни и лесни за навигиране елементи и обратна връзка към потребителя.

- **Уеб приложение**

Софтуерно приложение, достъпно чрез интернет браузър, работещо на уеб сървър и написано на езици като *HTML CSS*, *JavaScript* и сървърни езици като *PHP*, *Python* или *Ruby*. Позволява взаимодействие с потребителя и манипулиране на данни.

- **Потребител**

Човек, който използва уеб приложението или мрежова услуга.

- **Регистрация**

Процесът на предоставяне на лична информация за създаване на акаунт или профил, позволяващ достъп до услуга, система или приложение.

- **Логване**

Отнася се до процеса на въвеждане на потребителско име и парола (или други идентификационни данни), за да получите достъп до система, приложение или услуга. Влизането обикновено проверява самоличността на потребителя и му предоставя достъп до ресурсите или функциите, които са свързани с неговия акаунт.

- **Автентикация**

Процес на проверка на самоличността на потребител, устройство или система, като се използват идентификационни данни като потребителско име и парола.

- **Client**

Софтуерна програма или устройство, което се свързва със сървър и иска услуги или ресурси от него. Той управлява потребителския интерфейс и взаимодейства със сървъра. В архитектурата клиент-сървър клиентът е отговорен за логиката на представяне и потребителския интерфейс.

- **Сървър**

Компютърна програма или устройство, което предоставя услуги или ресурси на един или повече клиенти по мрежа. Сървърът обработва съхранението, управлението и обработката на данни.

- **База от данни**

Колекция от структурирани данни, която позволява лесно извличане, манипулиране и управление.

- **Client-Server**

Тип мрежова архитектура, при която сървър управлява и доставя централизиран ресурс или услуга, докато клиентите се свързват и изискват този ресурс или услуга през мрежа, използвайки протокол като TCP/IP. Той подобрява скалируемостта, сигурността и надеждността чрез разделяне на клиентската и сървърната логика.

2 ИЗПОЛЗВАНИ ТЕХНОЛОГИИ

1 Apache server (XAMPP)

Използва се като среда за разработка за уеб, което позволява тестването на кода локално, преди да се разположи на жив сървър. Освен това *XAMPP* улеснява настройването и управлението на локален уеб сървър.

2 HTML + CSS (+Bootstrap)

Потребителският интерфейс на системата е изграден предимно само с *HTML* и *CSS*. Всички страници и елементи като менюта, бутони и визуални обекти се създават без да се разчита на допълнителни библиотеки. Единственото изключение е страницата, която съдържа чат, която използва предварително създадени визуални ефекти от *Bootstrap*.

3 JS (client-side language) + jQuery (3.4.1)

Използван е за проверки на данни в отделните страници, както и за реализиране на клиентската част най-вече при *live chat*-а – клиентът е написан на *JS*, като се ползва *jQuery* за създаване, изпращане и получаване на заявки от клиента.

4 PHP (server-side language) + Ratchet Library (WebSockets)

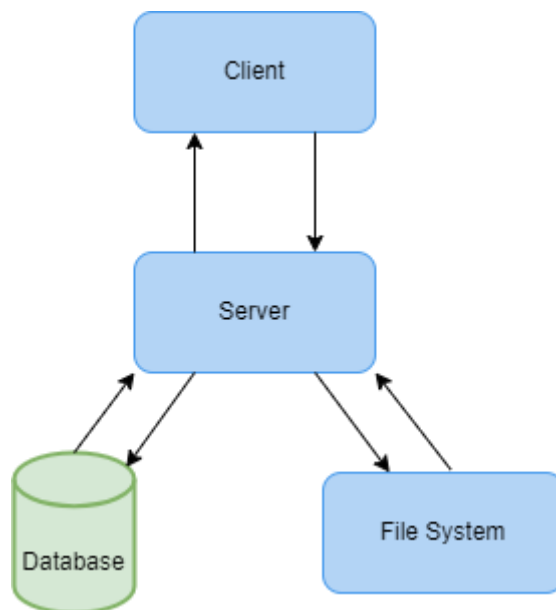
Страната на сървъра на системата е написана на *PHP* и библиотеката *Ratchet*, и по-конкретно – *WebSockets* за създаване и поддръжка на сокети. *Ratchet* се използва, защото осигурява най-добра производителност при използване на сокети за комуникация в реално време с *PHP*.

5 Composer

Composer е инструмент, който управлява зависимостите на *PHP* проект, включително библиотеки и пакети и техните версии. Използван е за инсталиране на *Ratchet* библиотеката и всички нейни зависимости на сървъра.

6 MySQL (XAMPP)

Базата данни на системата съхранява информация за потребителите, за албумите, които са качили, и за съобщенията в *live chat*-а.



- **Client**

Уеб браузърът използва *javascript* за предаване на данни към уеб сървър чрез *HTTP*. Потребителите могат да извършват различни действия като добавяне на албуми, редактиране на албуми, харесване на албуми, добавяне на описания и комуникация с други потребители в реално време. Въпреки това, преди да получат достъп до всички функции, потребителите трябва първо да попълнят формите за регистрация или вход.

- **Server**

Приложение, написано на *PHP*, приема заявки от клиенти. Сървърът поддържа удостоверяване и оторизация. Заявките се получават от клиентите, спрямо които се извършват съответните операции с данни.

- **Database**

Базата данни *MySQL* е достъпна от уеб сървъра и съхранява потребителска информация, като потребителски имена, пароли и имейли, в криптирана форма. Тя също така съдържа информация за албумите (за изображенията и за потребителите, които са ги публикували). Освен това базата данни съхранява информация за чата, включително всички минали съобщения, текущи потребители, времето на изпращане и подателя на всяко съобщение.

- **File System**

Файлова система, която съхранява всички изображения, добавени от потребителите в албуми, уеб сървърът отговаря за управлението на изображенията чрез добавяне, изтриване, редактиране след преглеждане и извършване на промени в базите данни.

3 РЕАЛИЗАЦИЯ НА БАЗАТА ОТ ДАННИ

Тъй като системата е реализирана с две основни функционалности

- 1) Създаване и качване на албуми от изображения и описания за тях
- 2) Възможност за комуникация между потребители в реално време (*Live chat*)

то сме създали две бази от данни, съответно базата данни *imagesdatabase* (в нея има информация само за албумите и всичко свързано с тях – път към дадения файл, описание на албум, заглавие, брой харесвания) и базата данни *websocket*, в която се намира информация за потребителите, които са налични на системата и информация за чата (съобщения и подробни детайли за тях).

В базата *imagesdatabase* се намира една таблица именувана *photos*, която има следната структура:

Таблица **photos**:

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
<input type="checkbox"/>	2	file_name	varchar(255)	utf8_bin		No	None			Change Drop More
<input type="checkbox"/>	3	user_name	varchar(100)	utf8_bin		No	None			Change Drop More
<input type="checkbox"/>	4	uploaded_on	datetime			No	None			Change Drop More
<input type="checkbox"/>	5	status	enum('1', '0')	utf8_bin		No	1			Change Drop More
<input type="checkbox"/>	6	text	mediumtext	utf8_bin		No	None			Change Drop More
<input type="checkbox"/>	7	album	varchar(100)	utf8_bin		No	None			Change Drop More
<input type="checkbox"/>	8	likesUsers	mediumtext	utf8_bin		No	None			Change Drop More
<input type="checkbox"/>	9	likesCount	int(11)			No	0			Change Drop More

- *file_name* - името на изображението, което се качва
- *user_name* - името на свързания с изображението/албума потребител
- *uploaded_on* - датата и часът, в който е качен албумът, или, ако е добавено изображение при редактиране на албум, съответно кога е добавено
- *text* - представлява текстовото описание към албума
- *album* - името на албума
- *likesUsers* - се ползват за следене на кой е харесал даден албум (за да не се случва даден потребител да може да харесва един албум повече от веднъж)

- *likesCount* съответно е за броя харесвания, за да могат да се представят визуално за всеки потребител, разглеждащ дадения албум

В базата данни *websocket* има 2 таблици със следните структури:

Таблица **users**:

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
<input type="checkbox"/>	2	username	varchar(100)	utf8_bin		No	None			Change Drop More
<input type="checkbox"/>	3	email	varchar(200)	utf8_bin		No	None			Change Drop More
<input type="checkbox"/>	4	password	varchar(255)	utf8_bin		No	None			Change Drop More

- *id* - всеки потребител има автоматично конфигуриран номер, който е уникален (конфигурира се при регистрация на потребител)
- *username*, *email* - потребителско име и имейл, които също са уникални за всеки отделен потребител – също се задават при регистрация, но първо се осъществява проверка в базата данни дали вече няма потребител със същите данни.
- *password* - паролата се криптира и се запазва в базата, за избягване на злоупотреби.

Таблица **chatrooms**:

	#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	id	int(11)			No	None		AUTO_INCREMENT	Change Drop More
<input type="checkbox"/>	2	userid	int(11)			No	None			Change Drop More
<input type="checkbox"/>	3	msg	varchar(200)	utf8_bin		No	None			Change Drop More
<input type="checkbox"/>	4	created_on	datetime			No	None			Change Drop More

- *id* – стаята за чат съдържа автоматично конфигуриращо се *id*, с което се характеризира всяко едно ново съобщение (задава се при пращане на съобщение)
- *userid* - пояснява даденото съобщение от кого е пратено – препратка към *id* от таблицата *users*, за да може всяко едно съобщение да се свързва съответно със своя изпращач
- *msg* – представлява самото съобщение под текстов формат
- *created_on* - времето, в което съобщението е изпратено

Запазването на съобщенията в базата и индексването им подред

Дава възможност за пазене на историята. По този начин всеки нов потребител ще може да преглежда вече изпратените съобщения. Посредством пазенето на *userid* и времето на изпращане за всяко съобщение можем да визуализираме в чата името на потребителя, изпратил конкретното съобщение, както и точните дата и час.

4 РЕАЛИЗАЦИЯ НА БИЗНЕС ЛОГИКАТА

Файлът **dbConfig** съдържа настройките, необходими за свързване към базата данни с изображения. Файловете, намиращи се в папката **dbB**, съдържат данните за свързване към базите данни на потребители и чат стаи и установяват връзката към базите данни. Освен това потребителската и чат информацията се представят под формата на класове. След като изображенията бъдат качени в системата, те се записват в папката за качване.

След като изображенията бъдат качени в системата, те се записват в папката за **uploads**. Папката **bin** съдържа сървър, който обработва комуникацията на чата чрез **WebSockets**. Папката **src** има компонент от страна на клиента на архитектурата клиент-сървър, който управлява функционалността за чат.

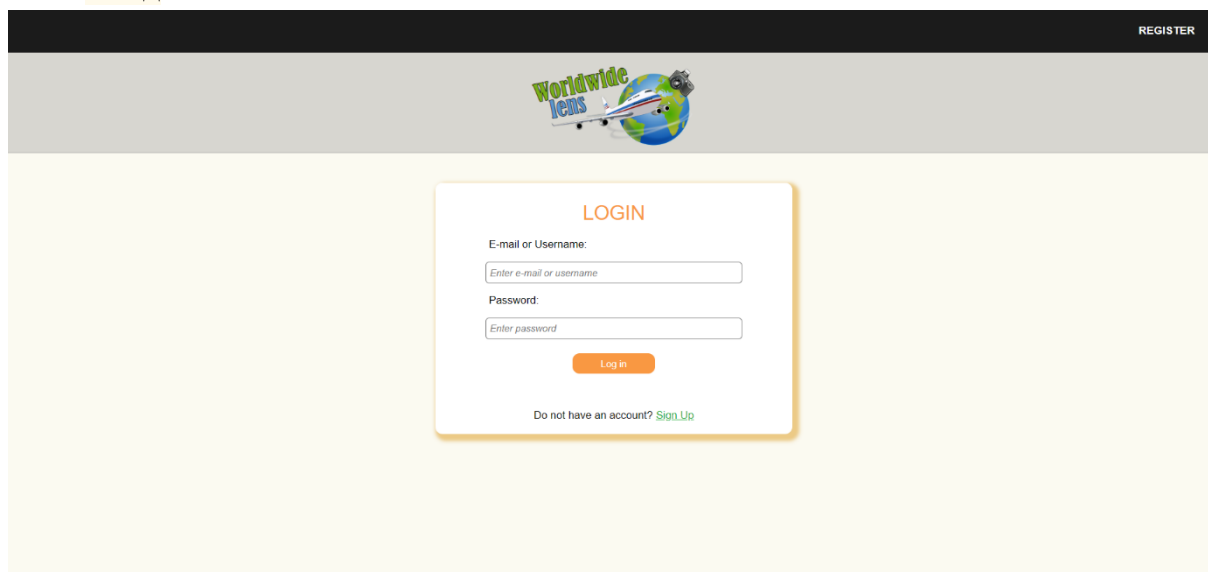
В папката **controllers** във файла **authController.php** изпълнява функционалностите за влизане/регистрация/излизане, както и валидирането на данните, въведени от съответните форми, когато потребителят натисне един от трите бутона. Файлът **upload.php** обработва основната функционалност за запазване на публикувани изображения в базата данни със снимки, като валидира данните, проверява дали нещо е добавено и дали е правилно прието от базата данни.

Файлът **index.php** показва потребителския интерфейс за основната страница на вече логнат потребител, който има възможност за преглед на вече качени албуми, както и достъп до формата за добавяне на албуми. Файлът **profile.php** позволява на потребителя да види своя собствен профил с всички албуми, добавени от него. В този случай, освен визуализация на албумите, има и форма, чрез която потребителят може да промени нещо по свой избор в даден албум. Файлът **delete.php** управлява промяната на състоянието на собствен албум (изтриването на албум, промяната на заглавието на албум, промяната на текстовата информация за албума и добавянето на нови изображения към избрания албум).

Файлът **likes.php** реализира функционалността на бутона за харесване на определен албум. Всеки потребител може да харесва даден албум, но може да даде само един глас. Ако потребителят вече е харесал албум, той може да го премахне своето харесване, като натисне отново бутона за харесване и вотът на албума ще бъде премахнат. Броят на харесванията и списъкът на потребителите, които са харесали даден албум, се съхраняват в базата данни. Всеки път, когато потребител хареса албум, списъкът се проверява, ако името му се съдържа, то се премахва от списъка и броят на харесванията се намалява с едно. Ако името на потребителя не е в списъка, гласът се увеличава с единица и името на потребителя се добавя към списъка.

5 РЕАЛИЗАЦИЯ НА ПОТРЕБИТЕЛСКИЯ ИНТЕРФЕЙС

5.1 Вход



The screenshot shows a web application interface for 'Worldwide Lens'. At the top, there is a dark header bar with a 'REGISTER' link on the right. Below the header is a light gray banner featuring the 'Worldwide Lens' logo, which includes a globe and an airplane. The main content area is a light yellow background. In the center, there is a white login form with an orange border. The form is titled 'LOGIN' in orange. It contains two input fields: 'E-mail or Username:' with a placeholder 'Enter e-mail or username' and 'Password:' with a placeholder 'Enter password'. Below these fields is an orange 'Log in' button. At the bottom of the form, there is a link: 'Do not have an account? [Sign Up](#)'.

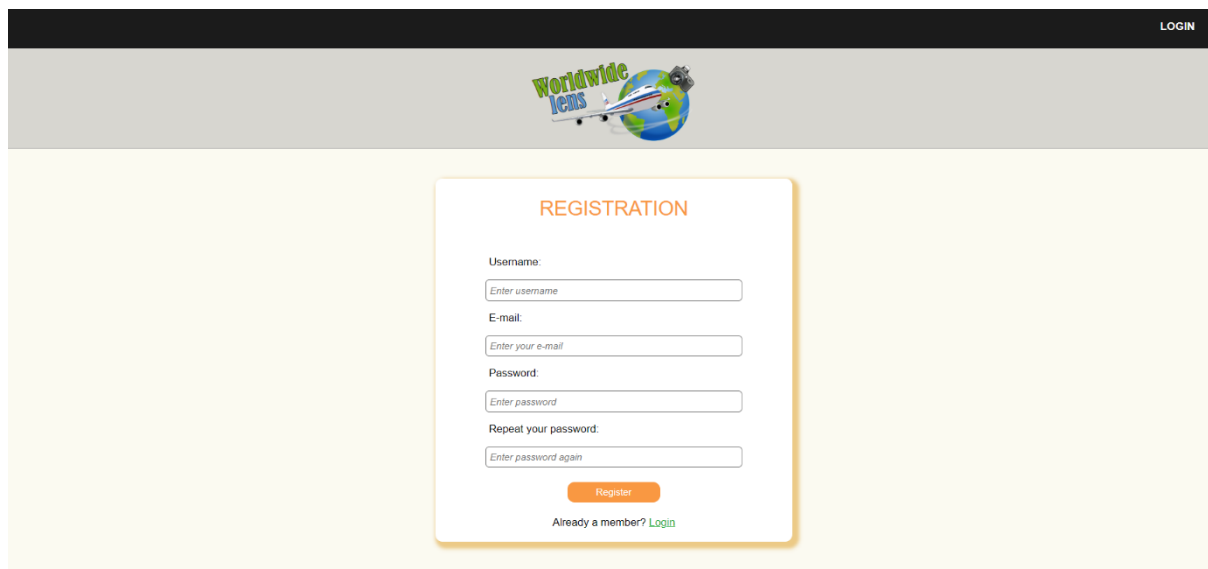
Достъпът до началната страница на системата е ограничен до регистрирани потребители, които са влезли. Ако не е установена връзка с акаунт, опитът за достъп до началната страница ще пренасочи потребителя към страницата за вход.

Страницата за вход разполага с формуляр в средата на страницата с две текстови полета, едното за въвеждане на имейл адрес или потребителско име, а другото за въвеждане на парола, и бутон за влизане под тях. След въвеждане на данни в двете полета и натискане на бутона, сървърът проверява базата данни за съвпадение между имейл и парола и ако има съвпадение, потребителят се логва успешно и се пренасочва към началната страница.

Ако някое от полетата е невалидно, се показва съобщение за грешка, което показва кое поле е неправилно и потребителят получава подкана да въведе нови валидни данни, ако иска да влезе в своя акаунт.

За потребители, които все още нямат акаунт, има бутони *Sign up* (във формуляра) или *Register* (в горния десен ъгъл на *header*-а), които пренасочват потребителя към страницата за регистрация, ако бъдат натиснати.

5.2 Регистрация

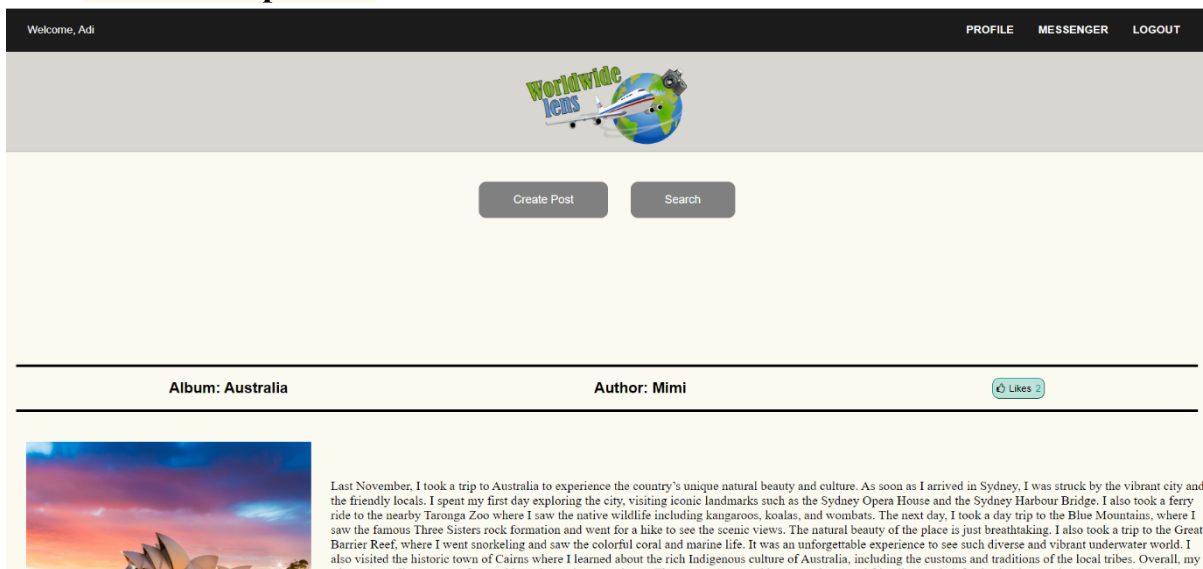


The screenshot shows a web page for 'Worldwide Lens' with a registration form. The page has a dark header with a 'LOGIN' link on the right. Below the header is a light gray banner featuring the 'Worldwide Lens' logo, which includes a globe and an airplane. The main content area is a light yellow background. In the center is a white registration form with an orange border. The form is titled 'REGISTRATION' in orange. It contains five input fields: 'Username' (placeholder: 'Enter username'), 'E-mail' (placeholder: 'Enter your e-mail'), 'Password' (placeholder: 'Enter password'), and 'Repeat your password' (placeholder: 'Enter password again'). Below the fields is an orange 'Register' button. At the bottom of the form, it says 'Already a member? [Login](#)'.

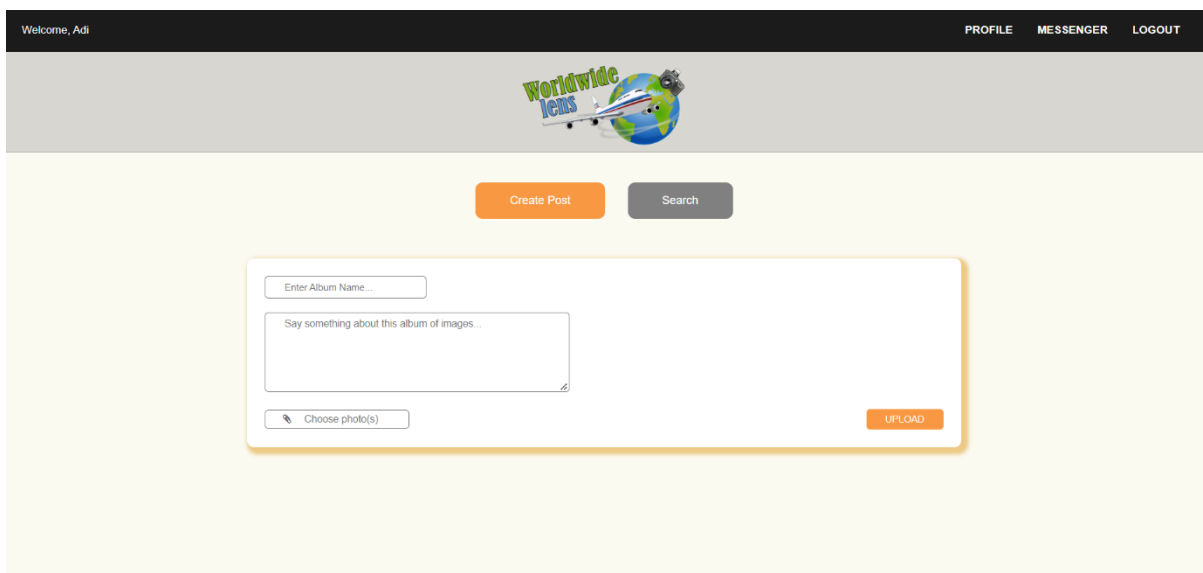
Страницата за регистрация разполага с формуляр в средата на страницата с 4 полета за въвеждане: уникално потребителско име, уникален имейл адрес, поле за парола и поле за повторение на парола. Ако потребител по погрешка отиде на тази страница, тъй като вече има акаунт от предишно използване на системата, той може лесно да бъде пренасочен към страницата за вход чрез натискане на бутоните *Login*, намиращи се под регистрационния формуляр и в горния десен ъгъл на *header*-а.

Подобно на страницата за вход, когато потребителят въведе данни в полетата и натисне бутона за регистрация, уеб сървърът проверява базата данни, за да се увери, че въведените данни вече не съществуват. Ако въведеният имейл или потребителско име вече се използва, ще се покаже съобщение за грешка, указващо това, и потребителят ще бъде подканен да въведе нови валидни данни. Извършва се и проверка, за да се гарантира, че двете полета за парола съвпадат. Ако всички данни са валидни, уеб сървърът ги криптира и ги записва в базата данни, като добавя нов потребител към таблицата с потребители в базата данни и след това пренасочва потребителя към началната страница на системата.

5.3 Начална страница



В горния ляв ъгъл се изписва приветстващо съобщение с потребителското име на текущо ползващия системата потребител. В горния десен ъгъл има три основни бутона, представляващи меню за избор – *Profile* е страницата, в която всеки потребител може да прегледа и редактира всички, публикувани от него албуми, *Messenger* представлява *Live chatroom*, в който всички потребители могат да контактуват в реално време един с друг, а при натискане на бутона *Logout*, потребителят напуска системата и сесията приключва (за да влезе отново, ще трябва да въвежда имейл и парола отново). Под логото в има 2 бутона – *Create a post* и *Search*.

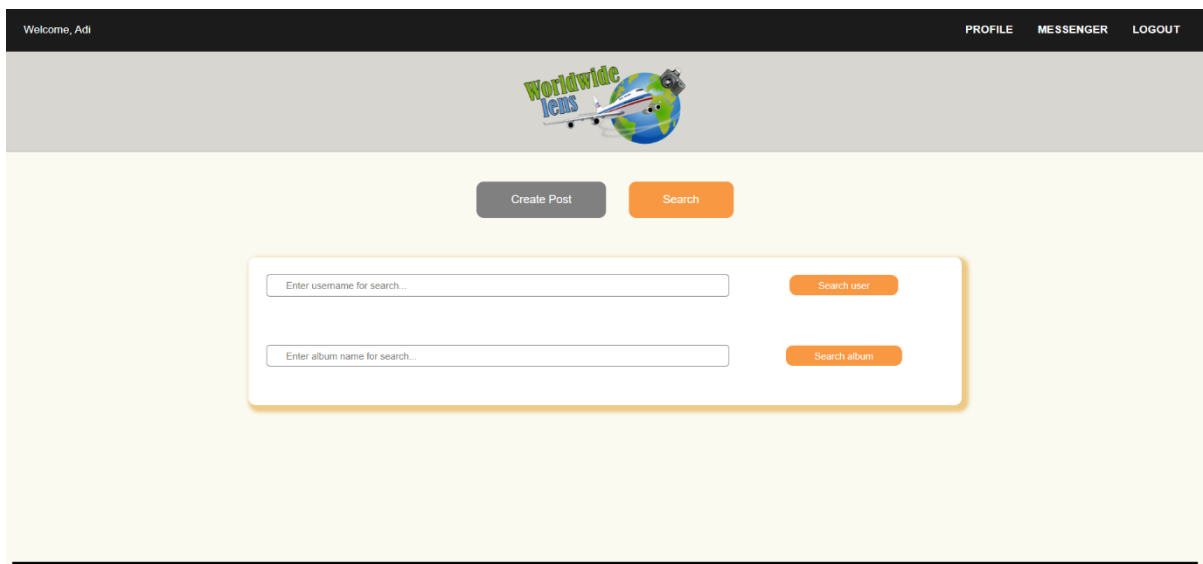


При натискане на бутона *Create a post* се отваря формата за създаване на публикация, съдържащо полета за заглавие на албум (име на посетената дестинация), описание на преживяването, бутон за добавяне на снимки и бутон за качване на публикацията.

При натискане на бутона за избор на файлове, потребителят може да избере едно или повече изображения. Поддържат се само основните формати *.jpg .bmp .png .jpeg* и има лимит за

размера на файловете, които се качват. В случай, че форматът на избраните файлове е друг или не е спазен размера на качването, албум не се създава, промените се изтриват.

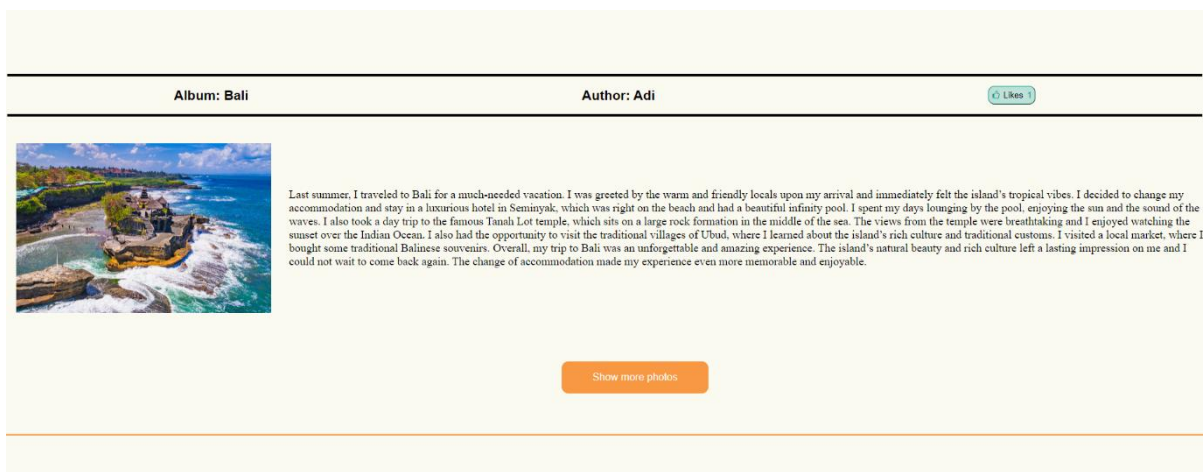
С натискането на бутона *Upload* се извършва проверка дали всички данни са коректни и дали няма празни полета, и ако няма върнато съобщение за грешка, то албумът се създава, страницата се презарежда и албумът се визуализира заедно с всичката информация за него.



При натискане на бутона *Search* се отваря формата за търсене съдържаща две текстови полета и два бутона.

В първото текстово поле при въвеждане на символен низ и натискане на бутона *Search user*, потребителят бива препратен към страница, в която са заредени всички албуми на потребител, чието потребителско име съдържа част или целия символен низ.

Във второто текстово поле при въвеждане на символен низ и натискане на бутона *Search album*, потребителят бива препратен към страница, в която са заредени всички албуми, чието заглавие или име на албум съдържа в себе си въведения символен низ.

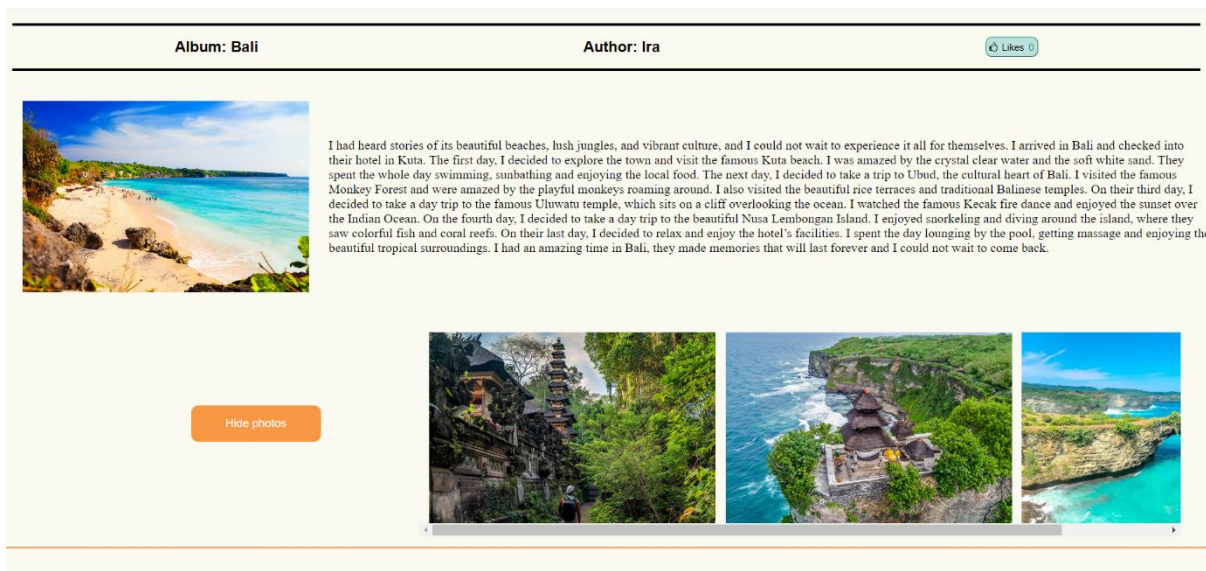


Album: Bali Author: Adi Likes 1

Last summer, I traveled to Bali for a much-needed vacation. I was greeted by the warm and friendly locals upon my arrival and immediately felt the island's tropical vibes. I decided to change my accommodation and stay in a luxurious hotel in Seminyak, which was right on the beach and had a beautiful infinity pool. I spent my days lounging by the pool, enjoying the sun and the sound of the waves. I also took a day trip to the famous Tanah Lot temple, which sits on a large rock formation in the middle of the sea. The views from the temple were breathtaking and I enjoyed watching the sunset over the Indian Ocean. I also had the opportunity to visit the traditional villages of Ubud, where I learned about the island's rich culture and traditional customs. I visited a local market, where I bought some traditional Balinese souvenirs. Overall, my trip to Bali was an unforgettable and amazing experience. The island's natural beauty and rich culture left a lasting impression on me and I could not wait to come back again. The change of accommodation made my experience even more memorable and enjoyable.

Show more photos

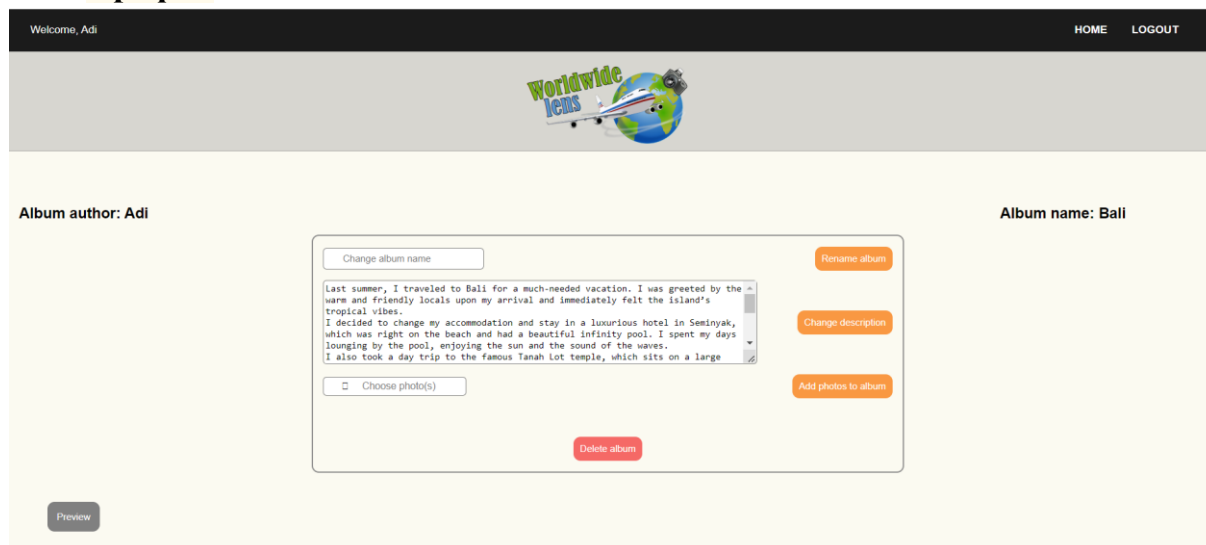
В началната страница албумите се визуализират един под друг като са ясно ограничени къде започват и свършват. Всеки албум започва с начална лента, в лявата страна на която се намира потребителското име на автора на албума, в средата на лентата стои името на дадения албум, а в дясната част – бутонът за харесване на албум и броят харесвания. След това се визуализира първата добавена снимка като главна и описанието, което потребителят е въвел. Под тях има бутон *Show more photos* и при натискането му се визуализират и останалите снимки от дадения албум, ако има такива.



Подобно на повечето социални мрежи всеки потребител може да харесва даден албум. Запазена е логиката, че ако даден потребител вече е харесал даден албум и отново натисне съответния бутон, то броят на харесванията намалява с едно, а в случай, че не го е харесал до момента, се увеличават с едно.

Подреждането на албумите се осъществява по азбучен ред, а в случай, че има няколко албума с едно и също име, но са на различни потребители, се подреждат по азбучен ред по името на потребителите. Системата не позволява един и същи потребител да публикува албуми с еднакви имена.

5.4 Профил



Следващата страница, до която регистриран потребител има достъп е *Profile* – всички албуми на текущо ползващия системата потребител.

Албумите отново са заредени в лексикографски ред по заглавие на албум.

Към всеки албум има формата с текстови полета и бутони, която дава възможност да бъде редактиран даден албум, да се добавят още снимки към него или да бъде изтрят.

При въвеждане на текст в текстовото поле за заглавие и натискане на бутона *Rename Album*, потребителят променя заглавието на албума, към който сочат текстовото поле и бутона.

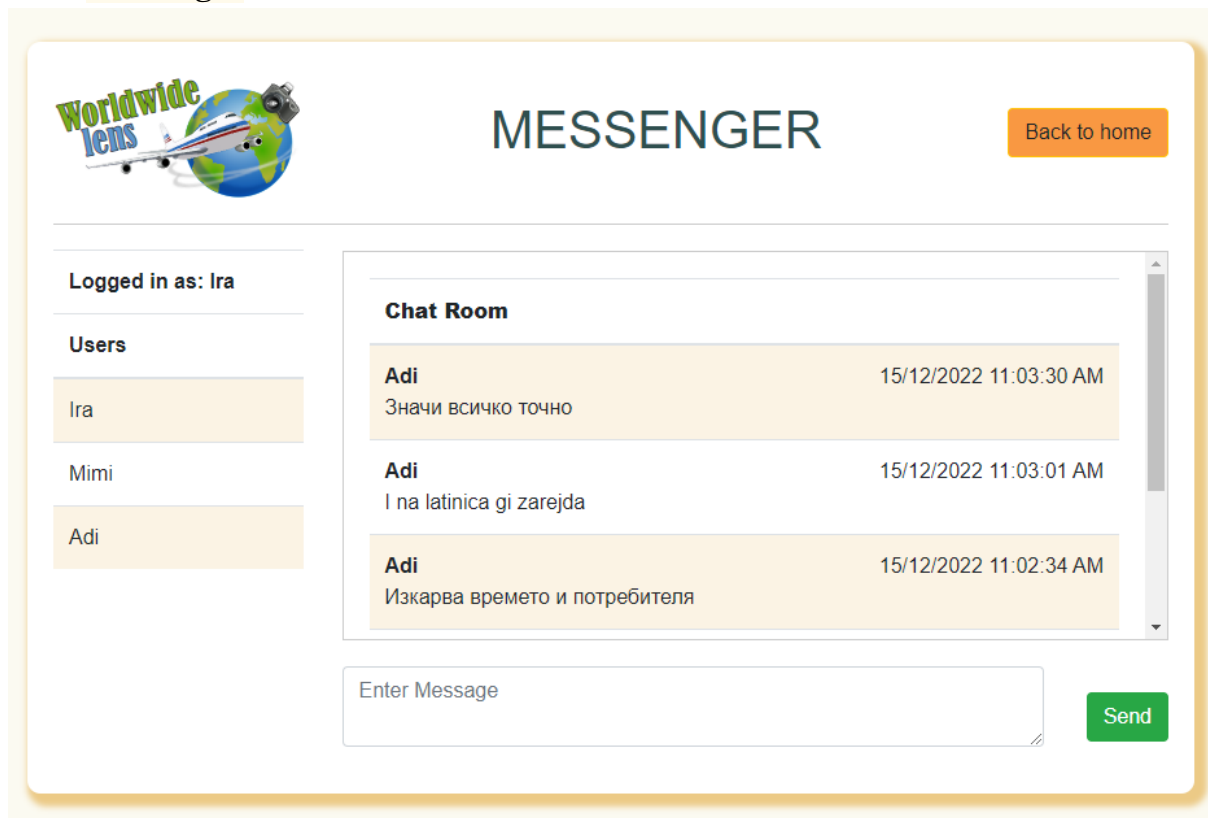
При въвеждане или пък изтриване на част от предишното описание от текстовото поле и натискане на бутона *Change Description*, потребителят променя описанието на съответния албум.

При натискане на бутона за избор на файлове и избиране на един или повече изображения и натискане на бутона *Update Album*, към албума, към който сочат бутоните биват добавени новоизбраните изображения.

При натискане на бутона *Delete Album*, потребителят изтрива албума, към който сочи бутона.

При натискане на бутона *Preview*, намиращ се в ляво под формата, потребителят може да види албума, по който прави редакции.

5.5 Messenger



Следващата страница е *Messenger – Chatroom*, в която потребителите контактуват едни с други в реално време без забавяне или рефрешване на страницата при пращане на съобщение.

Заглавната лента съдържа логото на системата, заглавието *MESSENGER*, както и бутон за връщане в началната страница.

Самият чат е разделен на две секции – в лявата под логото е ни показва потребителското име, с което сме логнати в момента, а под него всички потребители и техните имейли, които някога са влизали в системата. В дясната секция в прозореца *Chat Room* се изписва всяко едно съобщение в момента, в който бъде пратено. Над текста на съобщението в ляво стои потребителското име, което изпраща съобщението, а в дясно стои точния момент, в който е пратено съобщението. Под изпратените съобщения има текстово поле за въвеждане на съобщение и до него бутон за изпращане.

Всички съобщения се запазват в базата данни и затова дори при затваряне на чата, следващия път, когато го отворим, всички предходни съобщения няма да бъдат загубени. В случай на установяване на връзка със системата от даден потребител и пращане на съобщение в чата, то в сървърната част получаваме съобщения от следния вид:

New connection! (82)

Connection 82 sending message '{"userId":"6","msg":"test message"}' to 1 other connections

6 ВНЕДРЯВАНЕ НА СИСТЕМАТА

- 1 За да конфигурирате базите данни, *XAMPP* трябва да бъде инсталиран и *Apache Server* и *MySQL Admin* трябва да бъдат стартирани. Това ще създаде две бази данни: база данни с изображения - *imagesdatabase* и базата *websocket*.

Базата данни *imagesdatabase* ще има файл с конфигурация на структура на таблица и съответните примерни данни, поставени в нея, файлът се намира в **/newDB/photos.sql**, това е *sql* файл, който създава таблица със структура и примерни данни за изображения, той се намира в папката **newDB** на проекта.

Базата данни на *websocket* ще има два файла, поставени в нея: **/newDB/users.sql** и **/newDB/chatrooms.sql**, тези два файла са *sql* файлове, първият създава таблица със структура и примерни данни за потребителите, а вторият създава таблица със структура и примерни данни за чат със съобщения. И двата файла се намират в папката **newDB** на проекта.

- 2 След създаване на базите данни се пуска сървърът чрез терминал за *Windows* с командата:

PS C:\xampp\htdocs\web\bin> php server.php

Излиза съобщение, което индикира, че сървърът е стартирал – *Server started*.

- 3 Влизаме в браузър и въвеждаме следния адрес:

localhost/web/login.php

- 4 Въвеждаме някакви потребителски данни или се регистрираме. За улеснение в системата има 4 потребителски профила със следните данни:

Имейл	Парола
adi_dqksova@abv.bg	123456
irinka1212@abv.bg	123456
mimi.veleva1608@abv.bg	123456

- 5 За използване на чата натискаме бутона *Messenger* в заглавното меню на началната страница. Отваряме още един таб в режим, логваме се с друг потребителски акаунт и отново избираме *Messenger* в заглавното меню.
- 6 В случай, че чатът не работи, на вашата система трябва да инсталирате *Composer* <https://getcomposer.org/download/> - използва се за създаване на *dependencies* и изтегляне

на библиотеката, която сме ползвали за *WebSockets – Ratchet*. В папката, в която е проекта от командната линия въведете следния код:

C:\xampp\htdocs\web>composer require cboden/ratchet

Това ще инсталира библиотеката в папката. След това заменете съдържанието на *composer.json* със следното:

```
{
    "autoload": {
        "psr-4": {
            "MyApp\\": "src"
        }
    },
    "require": {
        "cboden/ratchet": "^0.4.1"
    }
}
```

От командната линия за папката, в която е проекта въведете:

C:\xampp\htdocs\web>composer update

Отново се върнете на точка 2 от описанието за внедряване на системата.

7 РАЗПРЕДЕЛЕНИЕ НА ДЕЙНОСТИТЕ ПО РЕАЛИЗАЦИЯТА

Структурата на базата данни беше обсъдена и създадена в началото от целия екип, като Ирина Христова се погрижи за попълване на нейното съдържание. По клиентската и сървърната част на системата се работеше паралелно от всички членове на екипа, като реализацията на отделните функционалности беше условно разделена по следния начин:

- Вход и регистрация – Ирина Христова;
- Начална страница (създаване на албум и преглед на всички качени в системата албуми) и страница с резултати от търсене на албум/потребител – Мария Велева;
- Профил (преглед, редакция и изтриване на собствени албуми) и създаване на чат за комуникация в реално време с използване сокети – Андрея Дякова.

В действителност всеки член от екипа помагаше на останалите, когато имаше такава нужда, и всички вземахме редовно участие при тестване на системата.